

EDA Case Study: Zomato Dataset

-Mayank Srivastava



This task will challenge your data analysis skills and give you a deeper understanding of the dataset. The Zomato Restaurants Dataset contains information about various restaurants, their ratings, cuisines, locations, and more. Your task is to perform EDA on this dataset and answer the following questions.

Dataset Link:https://drive.google.com/file/d/1r0rhEt3qdBys4MzdsLcUo76uWAwSGuD/view?usp=drive_link

Task instructions

1. Use Python and libraries like Pandas, Matplotlib, and Seaborn for data manipulation and visualization.
2. Before starting the analysis, clean the data by handling missing values and data types.
3. When answering each question, provide visualizations (such as bar plots, scatter plots, etc.) whenever applicable to support your findings.

```
In [1]: #importing important Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)

sns.set_style('darkgrid')
plt.style.use('ggplot')

In [2]: #importing the dataset
df=pd.read_csv('zomato.csv')
df.head()

Out[2]:   Unnamed: 0.1  Unnamed: 0  restaurant name  restaurant type  rate (out of 5)  num of ratings  avg cost (two people)  online_order  table booking  cuisines type  area  local address
0         0           0      #FeelTheROLL    Quick Bites        3.4             7          200.0       No        No      Fast Food  Bellandur  Bellandur
1         1           1        #L-81 Cafe    Quick Bites        3.9            48          400.0      Yes        No  Fast Food, Beverages  Byresandra,Tavarekere,Madiwala  HSR
2         2           2        #refuel        Cafe        3.7            37          400.0      Yes        No      Cafe, Beverages  Bannerghatta Road  Bannerghatta Road
3         3           3      '@ Biryani Central  Casual Dining        2.7           135          550.0      Yes        No      Biryani, Mughlai, Chinese  Marathahalli  Marathahalli
4         4           4      '@ The Bbq    Casual Dining        2.8            40          700.0      Yes        No  BBQ, Continental, North Indian, Chinese, Bever...  Bellandur  Bellandur

In [3]: #dropping unwanted columns
df.drop(['Unnamed: 0.1', 'Unnamed: 0'],axis= 1,inplace= True)

In [4]: #checking the shape
df.shape

Out[4]: (7105, 10)

In [5]: #checking the info
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7105 entries, 0 to 7104
Data columns (total 10 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   restaurant name    7105 non-null   object 
 1   restaurant type    7105 non-null   object 
 2   rate (out of 5)    7037 non-null   float64
 3   num of ratings     7105 non-null   int64  
 4   avg cost (two people) 7048 non-null   float64
 5   online_order       7105 non-null   object 
 6   table booking      7105 non-null   object 
 7   cuisines type      7105 non-null   object 
 8   area                7105 non-null   object 
 9   local address       7105 non-null   object 
dtypes: float64(2), int64(1), object(7)
memory usage: 555.2+ KB

In [6]: # checking for missing values
pd.DataFrame({'Missing val':df.isna().sum(), 'Missing val %':round(df.isna().sum()*100/len(df),2)})
```

Out[6]:

	Missing val	Missing val %
restaurant name	0	0.00
restaurant type	0	0.00
rate (out of 5)	68	0.96
num of ratings	0	0.00
avg cost (two people)	57	0.80
online_order	0	0.00
table booking	0	0.00
cuisines type	0	0.00
area	0	0.00
local address	0	0.00

In [7]: # making columns wise sumamry of values

```
for i in df.columns:  
    print(i)  
    print(df[i].unique())  
    print(df[i].nunique())  
    print('-'*50)
```

restaurant name
 [#FeelTheROLL' '#L-81 Cafe' '#refuel' ... "Zu's Doner Kebaps" 'Zyara'
 'Zyksha']
 7105

restaurant type
 ['Quick Bites' 'Cafe' 'Casual Dining' 'Takeaway, Delivery'
 'Bar, Casual Dining' 'Fine Dining' 'Pub' 'Casual Dining, Bar' 'Delivery'
 'Kiosk' 'Food Court' 'Microbrewery' 'Ban' 'Mess' 'Bakery'
 'Dessert Parlor' 'Beverage Shop' 'Sweet Shop' 'Dhaba' 'Cafe, Quick Bites'
 'Microbrewery, Casual Dining' 'Dessert Parlor, Beverage Shop'
 'Bar, Lounge' 'Quick Bites, Mess' 'Food Truck' 'Lounge, Bar'
 'Cafe, Dessert Parlor' 'Quick Bites, Beverage Shop' 'Pub, Casual Dining'
 'Pub, Microbrewery' 'Sweet Shop, Quick Bites' 'Cafe, Bakery'
 'Fine Dining, Bar' 'Bar, Pub' 'Bakery, Quick Bites'
 'Bakery, Beverage Shop' 'Bakery, Dessert Parlor' 'Takeaway' 'Lounge'
 'Microbrewery, Lounge' 'Lounge, Microbrewery'
 'Casual Dining, Microbrewery' 'Casual Dining, Pub' 'Bakery, Cafe'
 'Casual Dining, Cafe' 'Microbrewery, Pub' 'Cafe, Bar' 'Quick Bites, Cafe'
 'Cafe, Casual Dining' 'Pub, Bar' 'Quick Bites, Dessert Parlor'
 'Dessert Parlor, Quick Bites' 'Beverage Shop, Quick Bites'
 'Food Court, Quick Bites' 'Quick Bites, Sweet Shop'
 'Casual Dining, Lounge' 'Dessert Parlor, Bakery'
 'Dessert Parlor, Sweet Shop' 'Beverage Shop, Dessert Parlor'
 'Cafe, Lounge' 'Casual Dining, Quick Bites' 'Club'
 'Food Court, Casual Dining' 'Quick Bites, Bakery' 'Lounge, Casual Dining'
 'Food Court, Dessert Parlor' 'Dessert Parlor, Cafe' 'Lounge, Cafe'
 'Dessert Parlor, Food Court' 'Microbrewery, Bar' 'Confectionery'
 'Fine Dining, Lounge' 'Casual Dining, Irani Cafee'
 'Fine Dining, Microbrewery' 'Beverage Shop, Cafe' 'Bhojanalya'
 'Food Court, Beverage Shop' 'Quick Bites, Food Court'
 'Casual Dining, Sweet Shop' 'Quick Bites, Meat Shop'
 'Club, Casual Dining']

81

rate (out of 5)
 [3.4 3.9 3.7 2.7 2.8 4.1 3.2 3.5 4.4 4.2 4. 4.3 3.1 3.6 3.3 2.9 3. 3.8
 2.5 nan 4.6 1.8 2.4 4.5 4.9 2.1 4.7 2.6 4.8 2. 2.3 2.2]

31

num of ratings
 [7 48 37 135 40 305 49 41 22 2861 353 119
 1731 1509 218 24 620 58 33 595 11 209 419 932
 15 46 4 9 344 268 1234 283 21 128 156 556
 13 29 1149 662 17 14 399 55 82 23 62 737
 1763 51 478 8 531 18 5 16 105 74 106 101
 32 458 12 211 19 2871 6 10 25 544 97 42
 93 68 65 888 66 127 212 249 35 120 131 574
 134 54 164 61 1582 485 88 178 27 59 30 377
 76 783 867 72 20 336 39 258 1413 567 90 370
 185 260 561 89 113 682 346 446 704 77 224 26
 98 34 63 387 314 821 229 140 263 73 107 43
 108 148 256 57 31 136 38 47 64 130 171 84
 568 435 81 99 250 284 241 45 60 1428 317 52
 92 121 165 739 96 264 28 434 50 461 730 118
 841 179 474 236 195 53 189 339 8375 177 237 126
 83 533 449 277 69 861 137 278 2178 80 91 208
 1229 577 116 506 184 610 734 95 176 429 504 123
 634 194 86 479 903 527 1383 170 112 532 1014 44
 477 612 2212 197 764 257 207 110 1003 1303 1316 592
 173 151 169 2445 198 842 4650 75 496 345 869 266
 367 56 415 456 391 174 541 36 114 186 142 154
 244 521 201 858 253 1899 124 238 859 426 854 225
 259 115 133 153 418 779 1550 196 132 320 746 6865
 555 214 9272 678 347 85 292 158 301 147 751 71
 87 298 627 70 1071 618 1845 400 150 149 318 228
 1214 733 935 514 505 491 330 1721 866 717 2355 2616
 181 1636 601 700 973 1187 248 282 5894 970 2329 4321
 1633 94 141 1142 3870 213 191 879 356 351 3125 155
 2448 1594 665 111 182 782 1345 16345 1989 1233 402 688
 721 117 145 216 261 500 559 144 388 160 200 239
 1048 1654 1859 513 563 583 275 373 431 424 719 157
 79 540 192 231 1423 233 284 254 193 980 600 341
 343 172 420 444 293 825 67 1275 997 166 78 342
 2233 960 199 395 125 280 289 319 4955 378 307 455
 2658 445 183 358 122 295 557 308 247 7544 1403 2339
 1269 331 589 272 417 234 364 466 3998 182 820 129
 219 139 2589 235 4498 306 232 507 202 334 206 325
 384 475 152 255 489 242 183 2487 909 3423 498 109
 361 1414 1287 463 1800 281 372 798 2316 758 1341 652
 161 297 163 371 276 2651 1445 274 187 291 327 3826
 497 2240 1531 287 3126 3197 608 669 2192 731 1095 1077
 1563 553 1065 221 2392 1272 2039 383 210 950 523 6348
 597 138 888 454 648 302 7154 3061 471 167 1386 1146
 366 379 702 551 407 414 286 1249 222 159 300 836
 707 794 777 469 175 1824 270 3447 732 2 2838 715
 564 321 637 332 827 815 1875 868 951 1189 2781 380
 1796 252 2151 180 633 5262 666 1365 368 880 1116 1580
 4767 7193 1783 787 485 328 3573 294 460 311 1463 357
 203 374 1369 617 1837 238 767 246 535 1218 674 2368
 864 323 1 310 675 775 3421 519 977 1042 2807 1827
 315 220 162 476 2111 528 285 451 570 217 543 889
 1555 363 885 355 630 988 248 488 215 800 398 412
 571 1697 3605 1901 396 326 1166 184 309 1381 404 1235
 580 593 687 428 251 464 786 3586 1024 1717 789 907
 146 668 223 860 1004 881 1778 337 1541 2558 406 560
 1320 288 546 430 397 340 375 1745 3334 785 599 2266
 190 468 1240 914 335 684 625 3848 547 382 2709 697
 727 1286 2031 465 296 1213 484 920 1282 1056 349 522
 273 386 393 376 2096 168 1070 411 671 1156 439 1105
 511 2194 539 2729 2062 1236 324 413 1496 1117 837 900
 1388 756 1075 243 490 929 1644 765 245 847 481 548
 1089 1133 486 2921 784 1021 552 650 823 322 7854 956
 673 416 1025 457 385 741 725 313 763 804 1980 501
 550 624 979 487 143 796 1877 750 1047 774 262 3056
 1499 1279 582 1372 667 813 518 2073 677 442 2044 918
 432 1028 421 1193 759 441 467 538 1101 392 267 271
 1867 2249 2486 1313 840 492 3083 1099 1554 1932 681 566
 537 1418 3116 953 1285 2498 3651 524 290 269 433 1033
 1236 755 558 226 269 2315 2022 3458 499 591 1647 651
 946 227 611 1360 1788 6745 619 1141 656 981 1337 4575
 1380 473 2283 520 10413 4187 4947 512 3077 1739 3768 4108
 1143 797 1060 7113 2418 793 773 1196 1408 3953 542 569
 2923 2270 672 2745 1573 1724 680 1318 483 2540 1585 1037
 516 768 628 1937 1852 985 450 720 1701 1113 2075 1012
 957 188 303 3667 882 2901 2182 1082 14956 1038 654 425
 3568 360 448 502 766 2878 6907 2407 1617 4460 1225 338
 676 1858 354 695 438 1774 2773 5890 1587 459 1052 408
 1533 2332 1777 939 694 1538 410 1336 586 510 894]

935

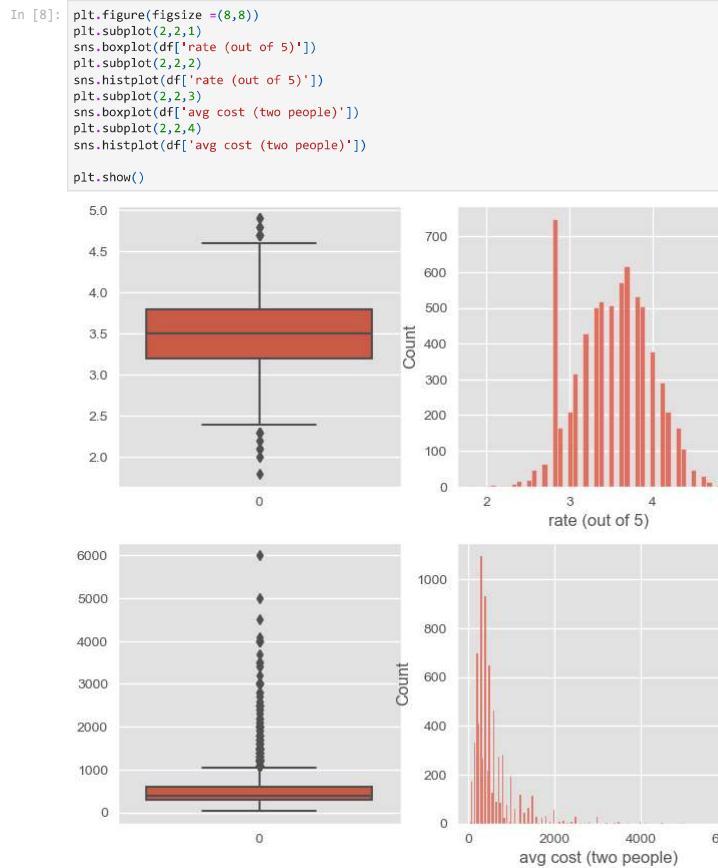
avg cost (two people)
 [200. 400. 550. 700. 300. 450. 1500. 2000. 800. 1400. 900. 600.
 1000. 2500. 3000. 250. 500. 1100. 1600. 150. 100. 1300. 350. 750.
 650. nan 1800. 4000. 1200. 1900. 160. 950. 3400. 850. 3700. 1700.
 80. 2400. 2200. 2100. 3200. 330. 360. 2800. 3500. 1250. 2600. 2700.
 1650. 4100. 6000. 120. 4500. 70. 240. 130. 1450. 2300. 230. 5000.
 50. 40. 560. 1850. 1350. 199.]

65

```

online_order
['No' 'Yes']
2
-----
table booking
['No' 'Yes']
2
-----
cuisines type
['Fast Food' 'Fast Food, Beverages' 'Cafe, Beverages' ...
'Italian, Mexican, Mediterranean, North Indian, Continental'
'Cafe, Italian, Continental, Burger'
'Turkish, Fast Food, Biryani, Chinese']
2175
-----
area
['Bellandur' 'Byresandra,Tavarekere,Madiwala' 'Bannerghatta Road'
'Marathahalli' 'Whitefield' 'Banashankari' 'Indiranagar'
'Electronic City' 'Old Airport Road' 'Basavanagudi' 'Malleshwaram'
'Kalyan Nagar' 'Rajajinagar' 'Brigade Road' 'Koramangala 7th Block'
'JP Nagar' 'Lavelle Road' 'Sarjapur Road' 'HSR' 'Frazer Town'
'Kammanahalli' 'New BEL Road' 'Jayanagar' 'Brookefield'
'Koramangala 5th Block' 'Koramangala 4th Block' 'Koramangala 6th Block'
'Church Street' 'MG Road' 'Residency Road']
30
-----
local address
['Bellandur' 'HSR' 'Bannerghatta Road' 'Marathahalli' 'Whitefield'
'Kumaraswamy Layout' 'Old Airport Road' 'Koramangala 5th Block' 'BTM'
'Electronic City' 'Indiranagar' 'Koramangala 3rd Block' 'JP Nagar'
'Malleshwaram' 'Kammanahalli' 'Rajajinagar' 'Basavanagudi'
'Infantry Road' 'Kalyan Nagar' 'Church Street' 'Jayanagar'
'Cunningham Road' 'Seshadripuram' 'Banaswadi' 'Kaggadasapura'
'Wilson Garden' 'Banashankari' 'Race Course Road' 'Koramangala 6th Block'
'CV Raman Nagar' 'Brigade Road' 'Nagawara' 'New BEL Road'
'Koramangala 1st Block' 'Brookefield' 'Jeevan Bhima Nagar' 'Hennur'
'Basaveshwara Nagar' 'Kanakapura Road' 'Yeshwantpur' 'Ejipura'
'Frazer Town' 'Koramangala 7th Block' 'Bommanahalli' 'Sarjapur Road'
'Thippasandra' 'Sanjay Nagar' 'Shanti Nagar' 'RT Nagar' 'Lavelle Road'
'Ulsoor' 'Vijay Nagar' 'Vasanth Nagar' 'Sadashiv Nagar' 'City Market'
'Shivajinagar' 'Magadi Road' 'Richmond Road' 'Residency Road'
'Koramangala' 'Domlur' 'East Bangalore' 'HBR Layout' 'Jalahalli'
'Majestic' 'St. Marks Road' 'MG Road' 'Rammurthy Nagar' 'KR Puram'
'South Bangalore' 'Koramangala 4th Block' 'Nagarbhavi'
'Commercial Street' 'Koramangala 8th Block' 'Sahakara Nagar' 'Hebbal'
'ITPL Main Road, Whitefield' 'Mysore Road' 'Uttarahalli'
'Varthur Main Road, Whitefield' 'Yelahanka' 'Sankey Road'
'North Bangalore' 'Old Madras Road' 'Hosur Road' 'Koramangala 2nd Block'
'West Bangalore' 'Peenya' 'Kengeri' 'Langford Town']
90
-----
```

Data Cleaning and Fixing missing values



In [9]: # due to outliers, for fixing missing values, we will use mean for rating and median for avg cost

Out[9]:

	rate (out of 5)	avg cost (two people)
mean	3.514253	540.286464
median	3.500000	400.000000

In [10]: df['rate (out of 5)'].fillna(df['rate (out of 5)'].mean(), inplace = True)
df['avg cost (two people)'].fillna(df['avg cost (two people)'].mean(), inplace = True)

Descriptive Statistics

In [11]: df.describe(include='all')

Out[11]:	restaurant name	restaurant type	rate (out of 5)	num of ratings	avg cost (two people)	online_order	table booking	cuisines type	area	local address
	count	7105	7105	7105.000000	7105.000000	7105	7105	7105	7105	7105
	unique	7105	81	NaN	NaN	NaN	2	2	30	90
	top	#FeelTheROLL	Quick Bites	NaN	NaN	NaN	Yes	No	North Indian, Chinese	Byresandra,Tavarekere,Madiwala
	freq	1	2840	NaN	NaN	NaN	3727	6361	421	798 459
	mean	NaN	NaN	3.514253	188.921042	540.286464	NaN	NaN	NaN	NaN
	std	NaN	NaN	0.461026	592.171049	461.041483	NaN	NaN	NaN	NaN
	min	NaN	NaN	1.800000	1.000000	40.000000	NaN	NaN	NaN	NaN
	25%	NaN	NaN	3.200000	16.000000	300.000000	NaN	NaN	NaN	NaN
	50%	NaN	NaN	3.500000	40.000000	400.000000	NaN	NaN	NaN	NaN
	75%	NaN	NaN	3.800000	128.000000	600.000000	NaN	NaN	NaN	NaN
	max	NaN	NaN	4.900000	16345.000000	6000.000000	NaN	NaN	NaN	NaN

Observations

- The Zomato Restaurants Dataset contains information of 7105 different Restaurants.
- There are 81 different Restaurant types, majority being 'Quick Bites' (2840 nos.)
- The average rating of the restaurants is 3.5
- The highest no. of ratings for a restaurant is 16,345 while the average no. of ratings is 190.
- The average cost of two people is ₹ 540 and goes as high as ₹ 6,000
- 3727/7105 restaurants provide the facility of online_order
- Table Booking facility is not available at 6361/7105 restaurants
- There Dataset covers restaurant data providing 2,175 different cuisines with most prominent being North Indian, Chinese.
- The dataset covers 30 different areas and 90 different local addresses.

1. What is the average rating of restaurants in the dataset?

In [12]: `round(df['rate (out of 5)'].mean(),1)`

Out[12]: 3.5

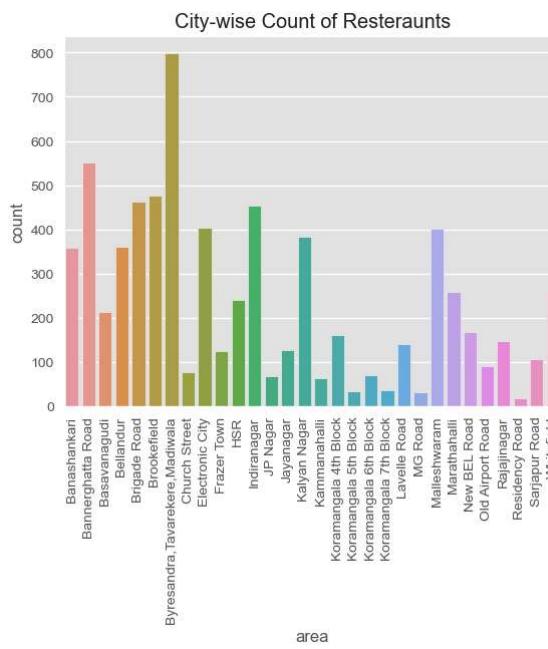
The average rating of restaurants in the dataset is 3.5

2. How many unique cities are represented in the dataset?

In [13]: `df.area.unique()`

Out[13]: 30

```
In [14]: x=df.area.unique()
x.sort()
sns.countplot(x=df.area, order = x)
plt.xticks(rotation= 90)
plt.title('City-wise Count of Resteraunts')
plt.show()
```

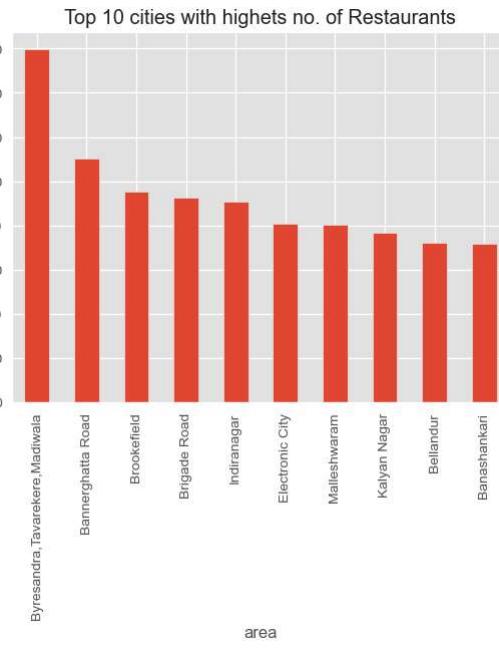


There are 30 unique cities in the dataset. Above is the countplot of city names sorted alphabetically.

3. Which city has the highest number of restaurants?

In [39]: `df.area.value_counts().head(10).plot(kind ='bar')`

```
plt.title('Top 10 cities with hights no. of Restaurants')
plt.show()
df.area.value_counts().head(10)
```



```
Out[39]: area
```

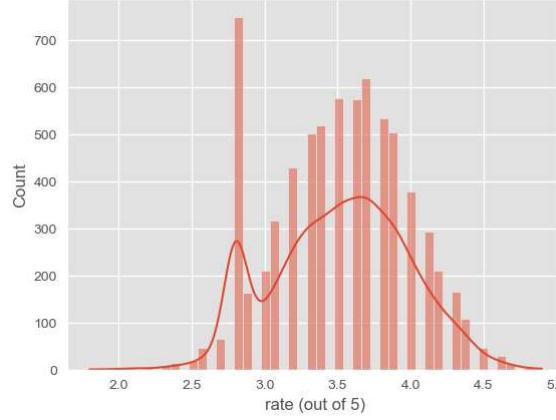
```
Byresandra,Tavarekere,Madiwala    798
Bannerghatta Road                 552
Brookefield                        477
Brigade Road                       464
Indiranagar                         455
Electronic City                     403
Malleshwaram                        402
Kalyan Nagar                        384
Bellandur                            361
Banashankari                        359
Name: count, dtype: int64
```

Above is the top 10 list of cities with highest no. of restaurants.

Byresandra,Tavarekere,Madiwala is at the top with 798 restaurants.

4. Plot the distribution of restaurant ratings.

```
In [16]: sns.histplot(x=df['rate (out of 5)'], kde = True)
plt.show()
```



```
In [17]: df[df['rate (out of 5)']<3.0]['rate (out of 5)'].mode()
```

```
Out[17]: 0    2.8
Name: rate (out of 5), dtype: float64
```

```
In [18]: df[df['rate (out of 5)']>3.0]['rate (out of 5)'].mode()
```

```
Out[18]: 0    3.7
Name: rate (out of 5), dtype: float64
```

The distribution of restaurant ratings is approximately a bell-shaped curve but has two peaks one at 2.8 and another at 3.7

5. Which cuisines are most common among the restaurants?

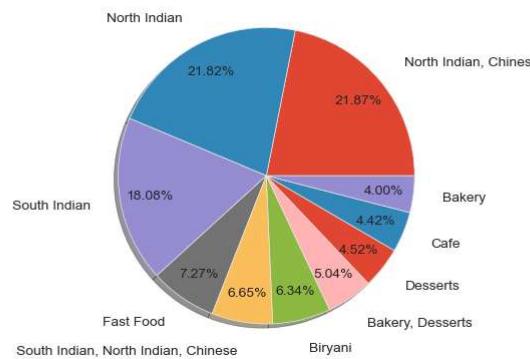
```
In [19]: df['cuisines type'].value_counts().head(10)
```

```
Out[19]: cuisines type
North Indian, Chinese            421
North Indian                      420
South Indian                      348
Fast Food                          140
South Indian, North Indian, Chinese 128
Biryani                           122
Bakery, Desserts                  97
Desserts                          87
Cafe                             85
Bakery                           77
Name: count, dtype: int64
```

There top 10 most common cuisines in the dataset are given above. We have used a pie-plot to understand the distribution.

```
In [20]: data=df['cuisines type'].value_counts().head(10)
plt.pie(data, labels = data.index, autopct='%.1f%%', shadow = True, pctdistance= 0.8, labeldistance=1.2)
plt.title('Pie chart for various Cuisine types available in the The Zomato Restaurants Dataset')
plt.show()
```

Pie chart for various Cuisine types available in the The Zomato Restaurants Dataset



6. What is the average cost for two people in different cities?

```
In [21]: round(df.groupby('area')['avg cost (two people)'].mean().sort_index(),2)
```

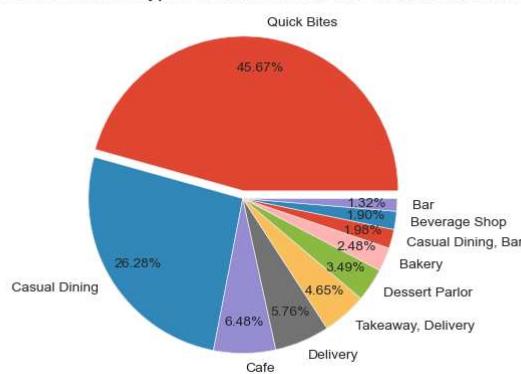
```
Out[21]: area
Banashankari          388.55
Bannerghatta Road     446.97
Basavanagudi           397.62
Bellandur              545.74
Brigade Road            836.27
Brookefield             526.83
Byresandra,Tavarekere,Madiwala 488.72
Church Street           747.28
Electronic City         512.38
Frazer Town             419.29
HSR                     460.54
Indiranagar             669.96
JP Nagar                441.30
Jayanagar               498.74
Kalyan Nagar             485.11
Kammanahalli            513.25
Koramangala 4th Block   568.34
Koramangala 5th Block   368.57
Koramangala 6th Block   403.11
Koramangala 7th Block   461.40
Lavelle Road             861.63
MG Road                  577.52
Malleshwaran             578.01
Marathahalli              520.89
New BEL Road              444.53
Old Airport Road          623.43
Rajajinagar               418.25
Residency Road            493.90
Sarjapur Road              475.84
Whitefield                 702.72
Name: avg cost (two people), dtype: float64
```

Above is an alphabetically sorted list of cities and their average cost for two people:

7. Visualize the distribution of restaurant types (e.g., Casual Dining, Café, etc.)

```
In [22]: plt.figure(figsize =(5,5))
data=df['restaurant type'].value_counts().head(10)
plt.pie(data, labels = data.index, autopct= '%1.2f%%', explode= [0.05,0,0,0,0,0,0,0,0,0],pctdistance=0.8)
plt.title('Pie chart for various Restaurant types available in the The Zomato Restaurants Dataset')
plt.show()
```

Pie chart for various Restaurant types available in the The Zomato Restaurants Dataset



```
In [23]: data
```

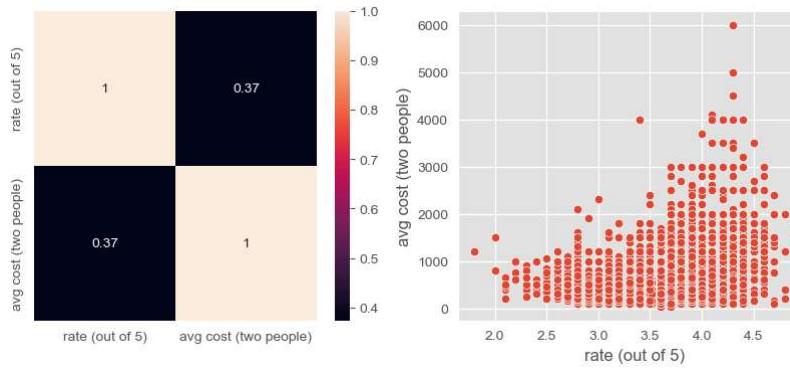
```
Out[23]: restaurant type
Quick Bites          2840
Casual Dining        1634
Cafe                 403
Delivery             358
Takeaway, Delivery    289
Dessert Parlor       217
Bakery               154
Casual Dining, Bar    123
Beverage Shop         118
Bar                   82
Name: count, dtype: int64
```

Above is the list of top 10 restaurant types in the dataset:

8. How does the rating correlate with the average cost for two people?

```
In [24]: plt.figure(figsize =(10,4))
corr=df[['rate (out of 5)', 'avg cost (two people)']].corr()
plt.subplot(1,2,1)
sns.heatmap(corr, annot= True)
plt.subplot(1,2,2)
sns.scatterplot(x='rate (out of 5)', y='avg cost (two people)', data=df)
```

```
Out[24]: <Axes: xlabel='rate (out of 5)', ylabel='avg cost (two people)'
```

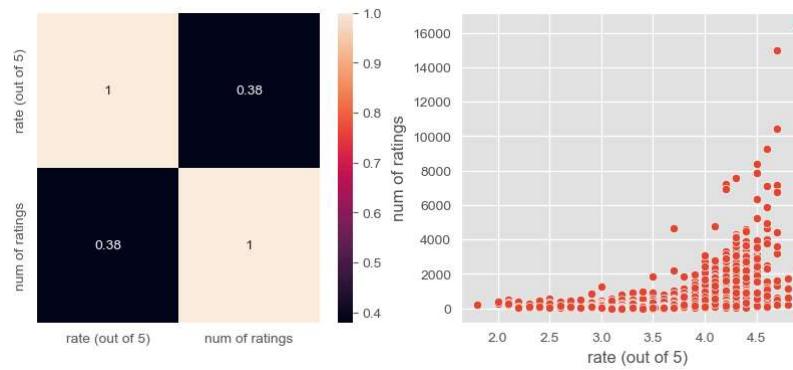


As the correlation coeff is <0.5, we can say that there is no correlation b/w Rating and avg cost for two

9. What is the correlation between the number of votes and the rating of restaurants?

```
In [25]: plt.figure(figsize =(10,4))
corr=df[['rate (out of 5)','num of ratings']].corr()
plt.subplot(1,2,1)
sns.heatmap(corr,annot= True)
plt.subplot(1,2,2)
sns.scatterplot(x='rate (out of 5)', y='num of ratings', data=df)
```

```
Out[25]: <Axes: xlabel='rate (out of 5)', ylabel='num of ratings'>
```



As the correlation coeff is <0.5, we can say that there is no correlation b/w Rating and No.of ratings

10. Which area has the highest number of highly-rated restaurants (rating > 4.5)?

```
In [26]: data=df[df['rate (out of 5)']>4.5]['restaurant name'].groupby(df.area).count()
print(f'Maximum of {data.max()} nos. of restaurant with rating >4.5 are preset at {data.idxmax()}' )
Maximum of 7 nos. of restaurant with rating >4.5 are preset at Byresandra,Tavarekere,Madiwala
```

```
In [27]: print(data.head(10).sort_values(ascending = False))
```

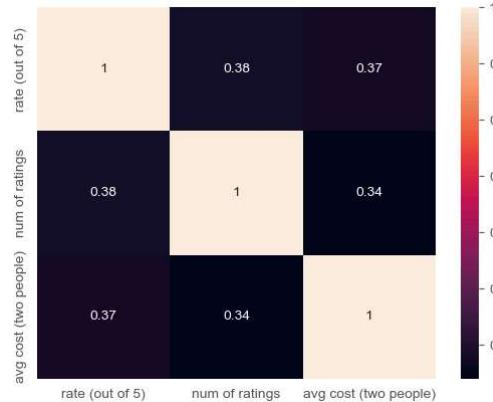
area	Count
Byresandra,Tavarekere,Madiwala	7
Brookefield	5
HSR	5
Indiranagar	5
Brigade Road	4
Banashankari	2
Bellandur	2
Jayanagar	2
Bannerghatta Road	1
Electronic City	1

Above is the list of areas with highest no. of highly rated restaurant (rating >4.5), with Byresandra,Tavarekere,Madiwala being on the top with 7 restaurants.

Some other observations

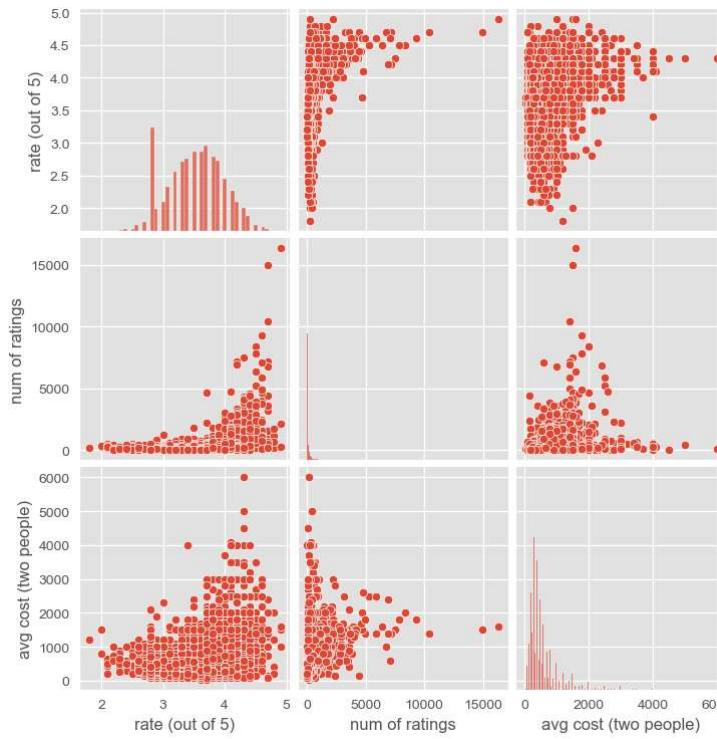
```
In [28]: # heatmap of all numerical columns
corr=df.corr(numeric_only = True)
sns.heatmap(corr, annot = True)
```

```
Out[28]: <Axes: >
```



```
In [37]: #pairplot
sns.pairplot(df)
```

```
Out[37]: <seaborn.axisgrid.PairGrid at 0x17a82157ad0>
```

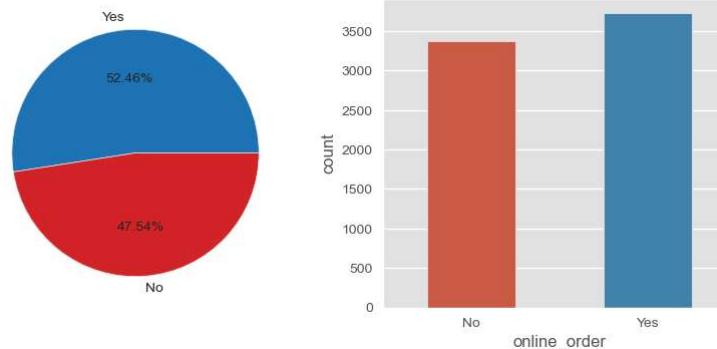


There is no correlation b/w ratings, avg cost of two and no. of ratings as all coefficients are <0.5

```
In [30]: # percent of restaurants with online order and table booking facility
plt.figure(figsize =(10,4))
```

```
plt.subplot(1,2,1)
data=df['online_order'].value_counts()
plt.pie(data, labels = data.index, autopct ='%1.2f%%', colors = ['tab:blue','tab:red'])
plt.subplot(1,2,2)
sns.countplot(x=df['online_order'], width =0.5)
```

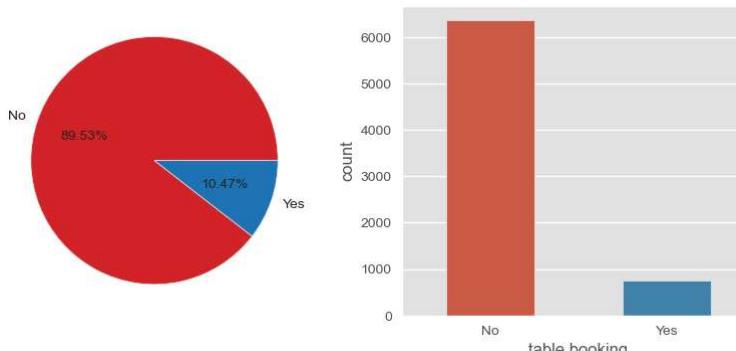
```
Out[30]: <Axes: xlabel='online_order', ylabel='count'>
```



```
In [31]: # percent of restaurants with online order and table booking facility
plt.figure(figsize =(10,4))
```

```
plt.subplot(1,2,1)
data=df['table booking'].value_counts()
plt.pie(data, labels = data.index, autopct ='%1.2f%%',colors = ['tab:red','tab:blue'])
plt.subplot(1,2,2)
sns.countplot(x=df['table booking'], width =0.5)
```

```
Out[31]: <Axes: xlabel='table booking', ylabel='count'>
```



```
In [32]: #lets find the most expensive and most cheap Restaurant in the dataset
```

```
df['avg cost (two people)'].agg(['min', 'max'])
```

```
Out[32]: min    40.0
max   6000.0
Name: avg cost (two people), dtype: float64
```

```
In [33]: df[df['avg cost (two people)']==6000]
```

Out[33]:

	restaurant name	restaurant type	rate (out of 5)	num of ratings	avg cost (two people)	online_order	table booking	cuisines type	area	local address
3561	Le Cirque Signature - The Leela Palace	Fine Dining	4.3	126	6000.0	No	Yes	French, Italian	Indiranagar	Old Airport Road

In [34]: df[df['avg cost (two people)']==40]

Out[34]:

	restaurant name	restaurant type	rate (out of 5)	num of ratings	avg cost (two people)	online_order	table booking	cuisines type	area	local address
5849	Srinidhi Sagar Deluxe	Quick Bites	3.6	108	40.0	Yes	No	South Indian, North Indian, Chinese	Koramangala 4th Block	Domlur
5850	Srinidhi Sagar Food Line	Quick Bites	3.7	129	40.0	Yes	No	South Indian, North Indian, Chinese	Brigade Road	Indiranagar

In [35]: # Lets find restaurants with Highest rating
df['rate (out of 5)'].agg(['max'])

Out[35]: max 4.9
Name: rate (out of 5), dtype: float64

In [36]: df[df['rate (out of 5)']==4.9]

Out[36]:

	restaurant name	restaurant type	rate (out of 5)	num of ratings	avg cost (two people)	online_order	table booking	cuisines type	area	local address
466	Asia Kitchen By Mainland China	Casual Dining, Bar	4.9	2178	1500.0	Yes	Yes	Asian, Chinese, Thai, Momos	Byresandra,Tavarekere,Madiwala	Koramangala 5th Block
1035	Byg Brewski Brewing Company	Microbrewery	4.9	16345	1600.0	Yes	Yes	Continental, North Indian, Italian, South Indi...	Bellandur	Sarjapur Road
5271	Sant...	Casual Dining	4.9	246	1000.0	No	Yes	Healthy Food, Salad, Mediterranean	Old Airport Road	Indiranagar

Observations:

1. The shape of the original dataset (7105, 12), but has two Unnamed cols. Useful data shape is (7105, 10)
2. The Zomato Restaurants Dataset contains information of 7105 different Restaurants.
3. There are 81 different Restaurant types, majority being 'Quick Bites' (2840 nos. i.e. 45.67%)
4. The average rating of the restaurants is 3.5
5. The highest no. of ratings for a restaurant is 16,345 while the average no. of ratings is 190.
6. The average cost of two people is ₹ 540 and goes as high as ₹ 6,000
7. 3727/7105 restaurants provide the facility of online_order
8. Table Booking facility is not available at 6361/7105 restaurants
9. The dataset covers restaurant data providing 2,175 different cuisines with most prominent being North Indian, Chinese.
10. The dataset covers 30 different areas and 90 different local addresses.
11. Byresandra,Tavarekere,Madiwala has the highest no. of restaurants (798 nos.).
12. The most expensive Restaurant is 'Le Cirque Signature - The Leela Palace' which is a Fine-dining place with French, Italian cuisine types, in Old Airport Road, Indiranagar. The place has no online order facility but table booking can be done.
13. Srinidhi Sagar Deluxe (Koramangala 4th Block) and Srinidhi Sagar Food Line (Brigade Road) are the restaurants with cheapest avg cost for two.
14. There are 3 Restaurants with Highest Rating of 4.9:
 - A. Asia Kitchen By Mainland China
 - B. Byg Brewski Brewing Company
 - C. Sant...
...
...
15. There is no correlation b/w ratings, no. of ratings and avg cost of two as all correlation coefficients are <0.5

-----End of Report-----